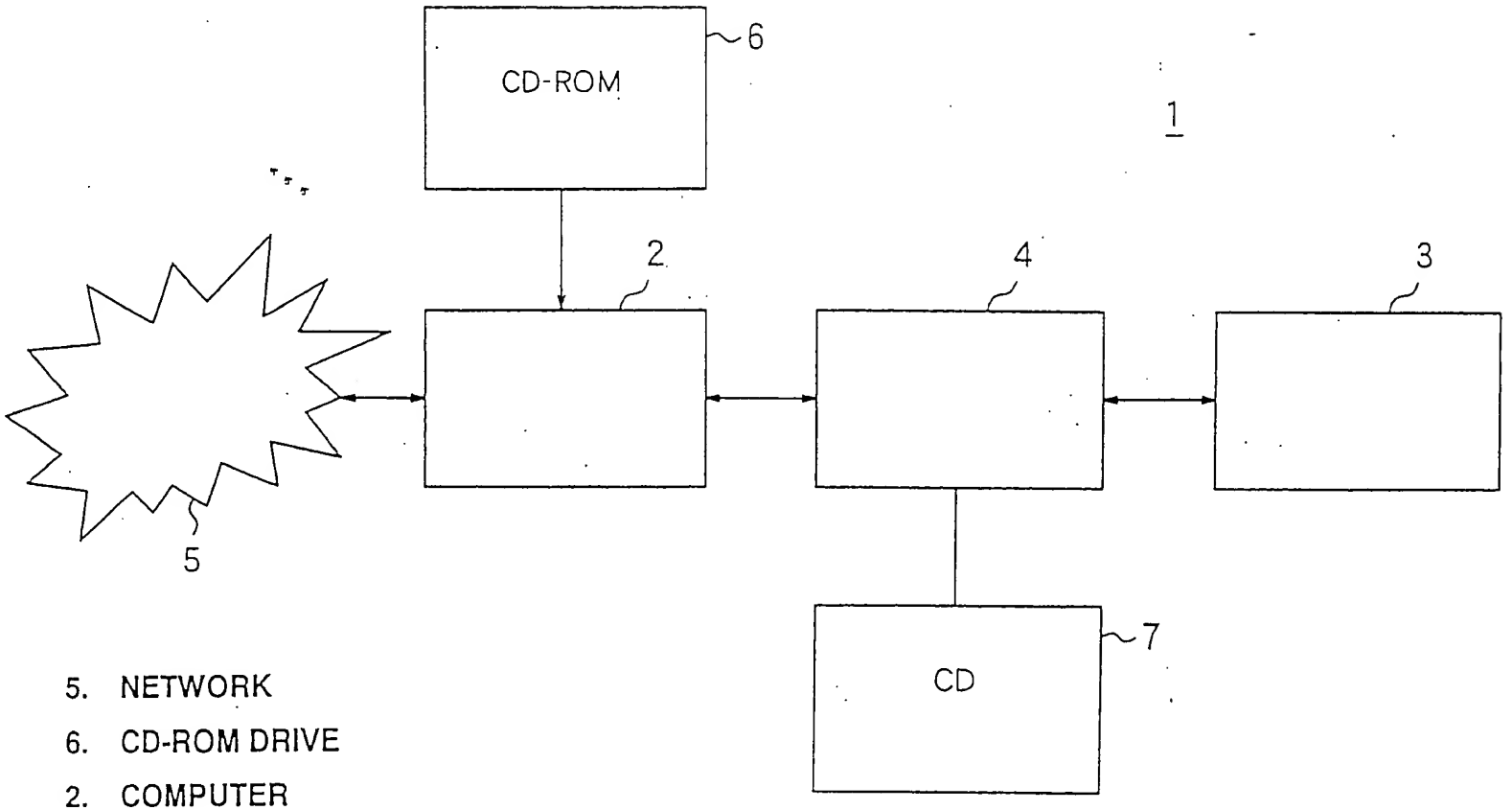
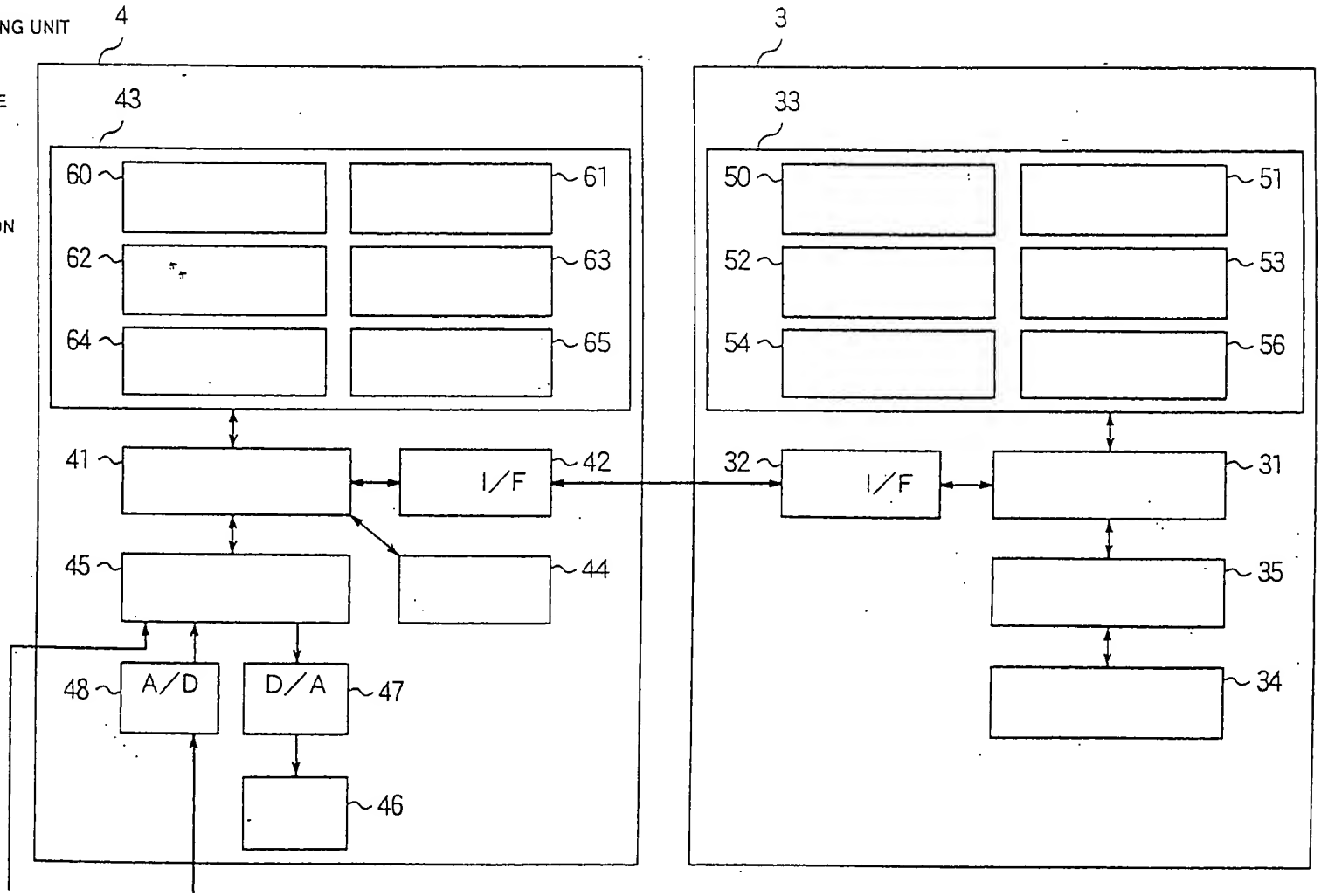



Figure 1

5. NETWORK
6. CD-ROM DRIVE
2. COMPUTER
4. PORTABLE PLAYER
7. CD PLAYER
3. PORTABLE STORAGE DEVICE (MEMORY STICK)

Figure 2




4. PORTABLE PLAYER
43. CONTROL MODULE
60. RANDOM NUMBER GENERATION UNIT

61. STORAGE UNIT
62. KEY CREATION/KEY PROCESSING UNIT
63. MUTUAL IDENTIFICATION UNIT
64. ENCRYPTING/DECRYPTING UNIT
65. CONTROL UNIT
41. MAIN CONTROL MODULE
42. COMMUNICATION I/F

15. COMPRESSION/EXPANSION MODULE

44. EDITING MODULE
48. A/D CONVERTER
47. D/A CONVERTER
46. SPEAKER

3. PORTABLE STORAGE DEVICE
33. CONTROL MODULE
50. RANDOM NUMBER GENERATION UNIT
51. STORAGE UNIT

AUDIO DATA (FROM COMPUTER 2)  AUDIO DATA (FROM CD PLAYER 7)

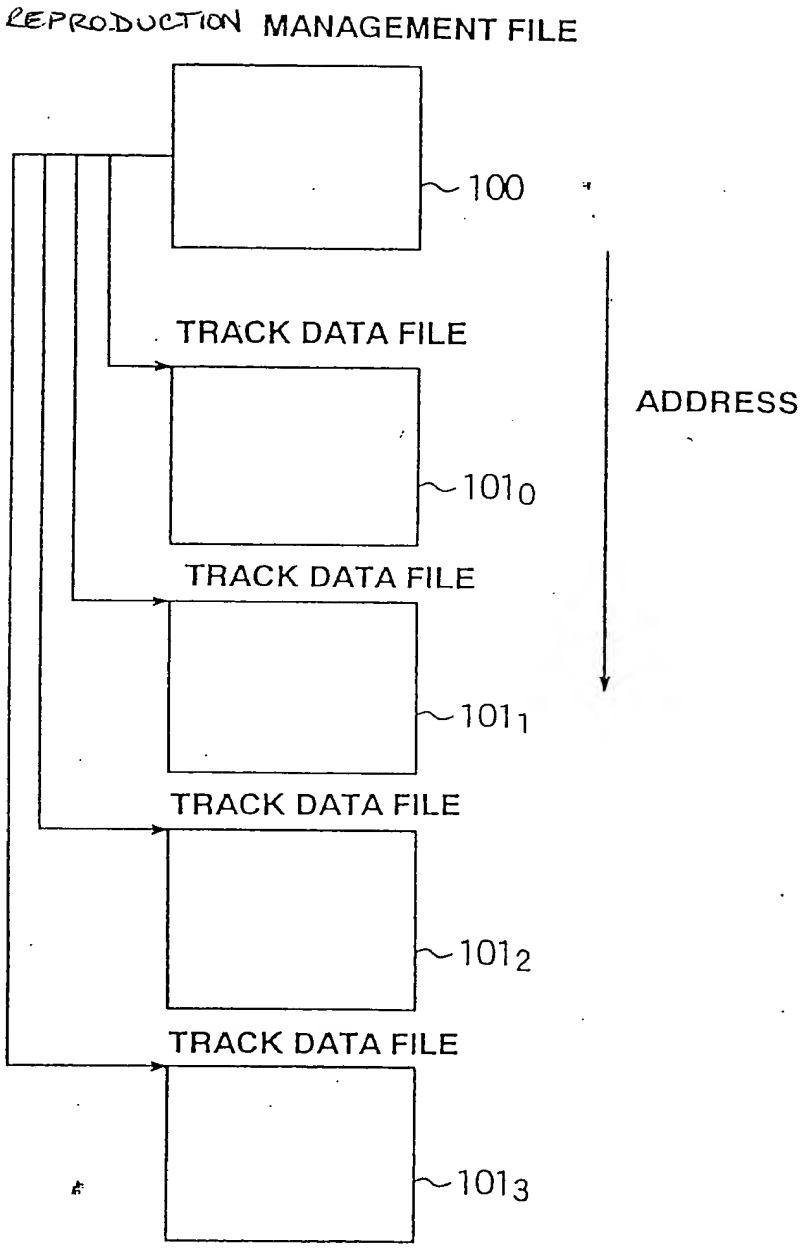
52. KEY CREATION/KEY PROCESSING UNIT
53. MUTUAL IDENTIFICATION UNIT
54. ENCRYPTING/DECRYPTING UNIT
55. CONTROL UNIT
32. COMMUNICATION I/F
31. MAIN CONTROL MODULE
35. FLASH MEMORY MANAGEMENT MODULE

Figure 3

DATA STORED IN STORAGE UNIT 51 OF PORTABLE STORAGE DEVICE 3

| | |
|---|---|
| IDENTIFICATION KEY DATA | $I K_0$ |
| | $I K_1$ |
| | $I K_2$ |
| | $I K_3$ |
| | ⋮ |
| | $I K_{30}$ |
| | $I K_{31}$ |
| DEVICE IDENTIFICATION DATA | $I D_m$ |
| STORAGE USE KEY DATA | $SK_m$ |

Figure 4




STORAGE DATA OF FLASH MEMORY 34 OF PORTABLE STORAGE DEVICE 3

Figure 5

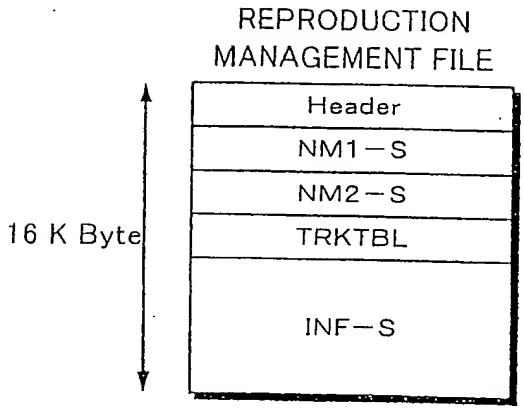

REPRODUCTION
MANAGEMENT FILE
Header
NM1—S
NM2—S
TRKTBL
INF—S
16 K Byte

Figure 6

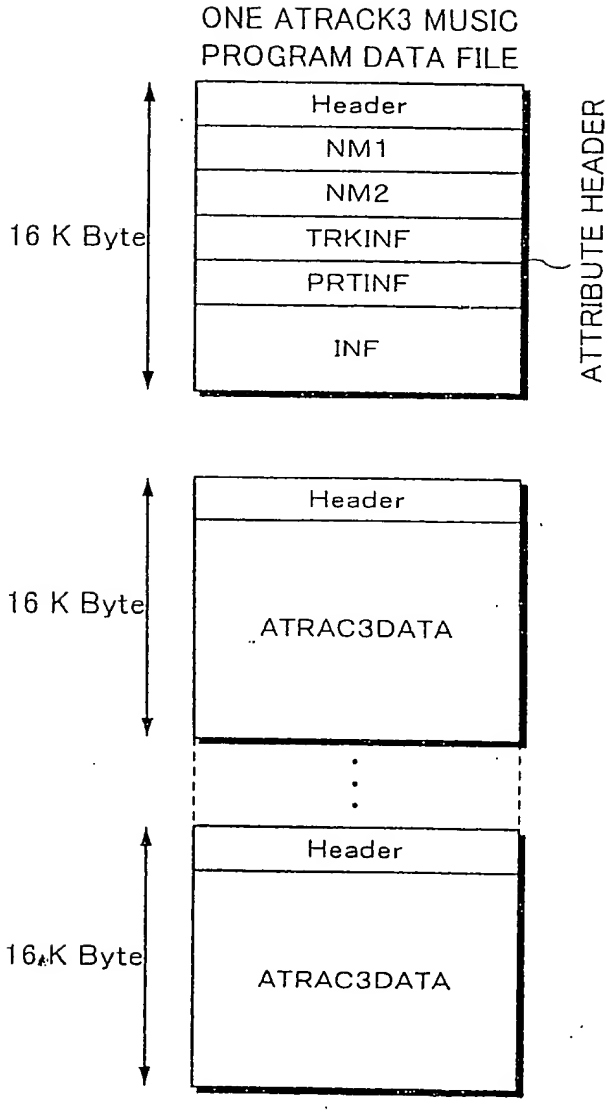

ONE ATRACK3 MUSIC
PROGRAM DATA FILE
Header
NM1
NM2
TRKINF
PRTINF
INF
ATTRIBUTE HEADER
16 K Byte
Header
ATRAC3DATA
16 K Byte
Header
ATRAC3DATA
16 K Byte

Figure 7

REPRODUCTION MANAGEMENT FILE

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0X0000 | BLKID−TL0 | | | | Reserved | | MCode | | REVISION | | | | Reserved | | | |
| 0X0010 | SN1C+L | | SN2C+L | | SINFSIZE | | T−TRK | | VerNo | | Reserved | | | | | |
| 0X0020 | NM1−S(256) | | | | | | | | | | | | | | | |
| 0X0120 | NM2−S(512) | | | | | | | | | | | | | | | |
| 0X0320 | Reserved | | | | | | | | CONTENTSKEY | | | | | | | |
| 0X0330 | Reserved | | | | | | | | MAC | | | | | | | |
| | Reserved | | | | | | | | | | | | S−YMDhms | | | |
| TRKTBL 0X0350 | TRK−001 | | TRK−002 | | TRK−003 | | TRK−004 | | TRK−005 | | TRK−006 | | TRK−007 | | TRK−008 | |
| | TRK−009 | | TRK−010 | | TRK−011 | | TRK−012 | | TRK−013 | | TRK−014 | | TRK−015 | | TRK−016 | |
| | | | | | | | | | | | | | | | | |
| 0X0660 | TRK−393 | | TRK−394 | | TRK−395 | | TRK−396 | | TRK−397 | | TRK−398 | | TRK−399 | | TRK−400 | |
| 0X0647 | INF−S(14720) | | | | | | | | | | | | | | | |
| 0X3FF0 | BLKID−TL0 | | | | Reserved | | MCode | | REVISION | | | | Reserved | | | |

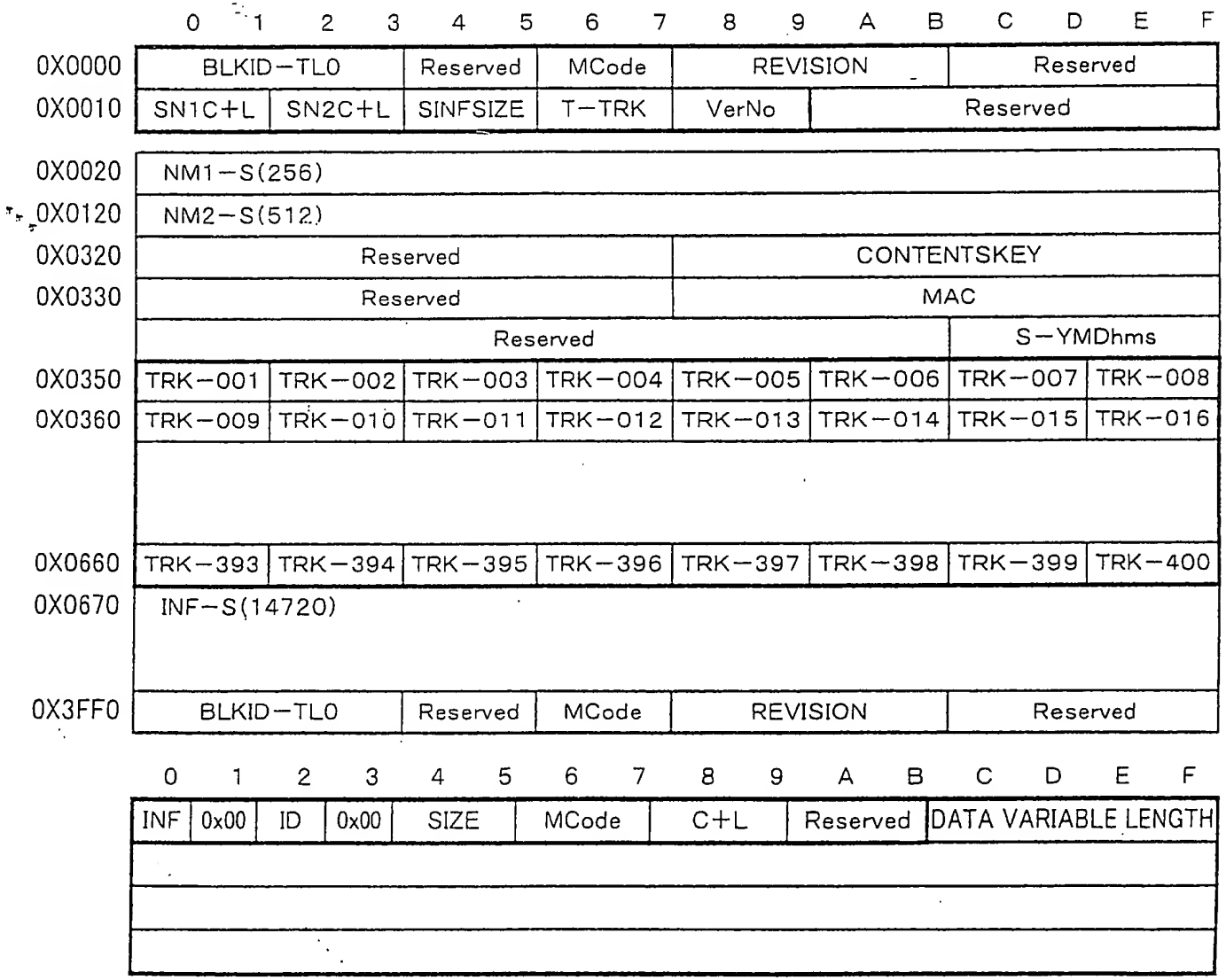

0 1 2 3 4 5 6 7 8 9 A B C D E F
0X0000
BLKID−TL0
Reserved
MCode
REVISION
Reserved
0X0010
SN1C+L
SN2C+L
SINFSIZE
T−TRK
VerNo
Reserved
0X0020
NM1−S(256)
0X0120
NM2−S(512)
0X0320
Reserved
CONTENTSKEY
0X0330
Reserved
MAC
Reserved
S−YMDhms
0X0350
TRK−001
TRK−002
TRK−003
TRK−004
TRK−005
TRK−006
TRK−007
TRK−008
0X0360
TRK−009
TRK−010
TRK−011
TRK−012
TRK−013
TRK−014
TRK−015
TRK−016
0X0660
TRK−393
TRK−394
TRK−395
TRK−396
TRK−397
TRK−398
TRK−399
TRK−400
0X0670
INF−S(14720)
0X3FF0
BLKID−TL0
Reserved
MCode
REVISION
Reserved
0 1 2 3 4 5 6 7 8 9 A B C D E F
INF
0x00
ID
0x00
SIZE
MCode
C+L
Reserved
DATA VARIABLE LENGTH

Figure 8

Figure 9

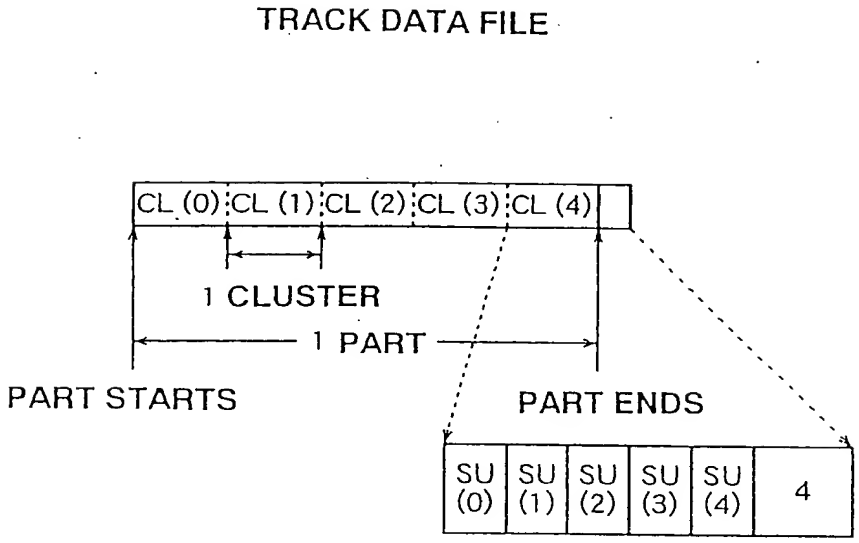

TRACK DATA FILE
CL (0)
CL (1)
CL (2)
CL (3)
CL (4)
1 CLUSTER
1 PART
PART STARTS
PART ENDS
SU (0)
SU (1)
SU (2)
SU (3)
SU (4)
4

Figure 10

A3Dnnnnn.MSA(ATRAC3 DATA FILE)

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x0000 | BLKID-HD0 | | | | Reserved | | MCode | | Reseved | | | | BLOCK SERIAL | | | |
| 0x0010 | N1C+L | | N2C+L | | INFSIZE | | T-PRT | | T-SU | | | | INX | | XT | |
| 0x0020 | NM1(256) | | | | | | | | | | | | | | | |
| 0x0120 | NM2(512) | | | | | | | | | | | | | | | |
| 0x0310 | | | | | | | | | | | | | | | | |
| 0x0320 | Reserved(8) | | | | | | | | CONTENTSKEY | | | | | | | |
| | Reserved(8) | | | | | | | | MAC | | | | | | | |
| | Reserved(12) | | | | | | | | | | | | A | LT | FNo | |
| | MG(D)SERIAL-nnn | | | | | | | | | | | | | | | |
| 0x0360 | CONNUM | | | | YMDhms-S | | | | YMDhms-E | | | | MT | CT | CC | CN |
| 0x0370 | PRTSIZE | | | | PRTKEY | | | | | | | | Reserved(8) | | | |
| 0x0380 | | | | | CONNUM0 | | | | PRTSIZE(0x0388) | | | | PRTKEY | | | |
| 0x0390 | | | | | Reserved(8) | | | | | | | | CONNUM0 | | | |
| | INF(0x0400) | | | | | | | | | | | | | | | |
| 0x3FFF | BLKID-HD0 | | | | Reserved | | MCode | | Reseved | | | | BLOCK SERIAL | | | |
| 0x4000 | BLKID-A3D | | | | Reserved | | MCode | | CONNUM0 | | | | BLOCK SERIAL | | | |
| 0x4010 | BLOCK SEED | | | | | | | | INITILIZATION VECTOR | | | | | | | |
| 0x4020 | SU-000(Nbyte=384byte) | | | | | | | | | | | | | | | |
| 0x41A0 | SU-001(Nbyte) | | | | | | | | | | | | | | | |
| 0x4320 | SU-002(Nbyte) | | | | | | | | | | | | | | | |
| 0x04A0 | SU-041(Nbyte) | | | | | | | | | | | | | | | |
| 0x7DA0 | | | | | | | | | | | | | | | | |
| 0x7F20 | Reserved(Nbyte=208byte) | | | | | | | | | | | | | | | |
| 0x7FF0 | BLOCK SEED | | | | | | | | | | | | | | | |
| | BLKID-A3D | | | | Reserved | | MCode | | CONNUM0 | | | | BLOCK SERIAL | | | |

Figure 11

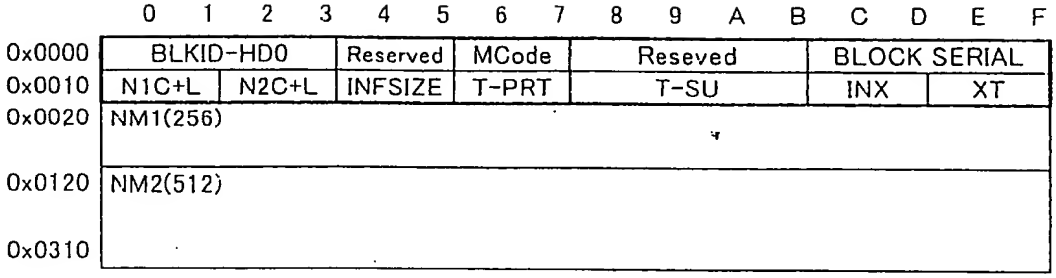



Figure 12

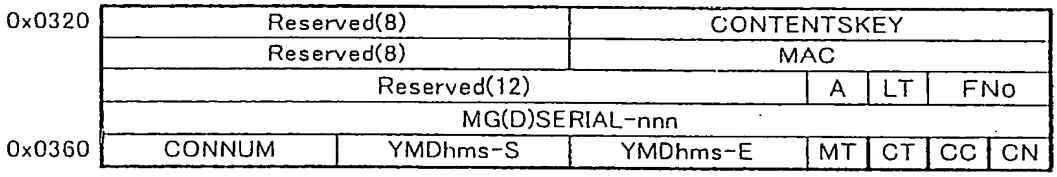



Figure 13

bit7:MODE OF ATRAC3 0:Dual 1:Joint

bit6,5,4 N OF 3 BITS:MODE VALUE

| N | MODE | TIME | TRANSMISSION RATE | SU | BYTES |
|---|---|---|---|---|---|
| 7 | HQ | 47min | 176kbps | 31SU | 512 |
| 6 | | 58min | 146kbps | 38SU | 424 |
| 5 | EX | 64min | 132kbps | 42SU | 384 |
| 4 | SP | 81min | 105kbps | 53SU | 304 |
| 3 | | 90min | 94kbps | 59SU | 272 |
| 2 | LP | 128min | 66kbps | 84SU | 192 |
| 1 | mono | 181min | 47kbps | 119SU | 136 |
| 0 | mono | 258min | 33kbps | 169SU | 96 |

bit3:Reserved

bit2:DATA TYPE 0:AUDIO 1:OTHER

bit1:REPRODUCTION SKIP 0:NORMAL REP 1:SKIP

bit0:EMPHASIS 0:OFF 1:ON(50/15 $\mu$ S)

Figure 14

| | | |
|---|---|---|
| | bit7 | :COPY PERMISSION 0:COPY PROHIBITION 1:COPY PERMISSION |
| | bit6 | :GENERATION 0:ORIGINAL 1:FIRST OR LATER COPY GENERATION |
| HCMS | bit5-4 | :COPY CONTROL FOR HIGH SPEED DIGITAL COPY |
| | | 00:COPY PROHIBITION 01:COPY FIRST GENERATION 10:COPY PERMISSION |
| | | COPY OF FIRST COPY GENERATION IS PROHIBITED. |
| | bit3-2 | MagicGate AUTHENTICATION LEVEL |
| | | 00:Level10(Non-MG) 01:Level1 |
| | | 10:Level2 11:Reserved |
| | | DIVIDE AND COMBINE ARE PROHIBITED IN OTHER THAN LEVEL 10. |
| | bit1,0 | Reserved |

Figure 15

| | | | | |
|---|---|---|---|---|
| 0x0370 | PRTSIZE | PRTKEY | | Reserved(8) |
| 0x0380 | | CONNUM0 | PRTSIZE(0x0388) | PRTKEY |
| 0x0390 | | Reserved(8) | | CONNUM0 |

Figure 16

| | | | | | |
|---|---|---|---|---|---|
| 0x4000 | BLKID-A3D | Reserved | MCode | CONNUM0 | BLOCK SERIAL |
| 0x4010 | BLOCK SEED | | | INITILIZATION VECTOR | |
| 0x4020 | SU-000(Nbyte=384byte) | | | | |

Figure 17

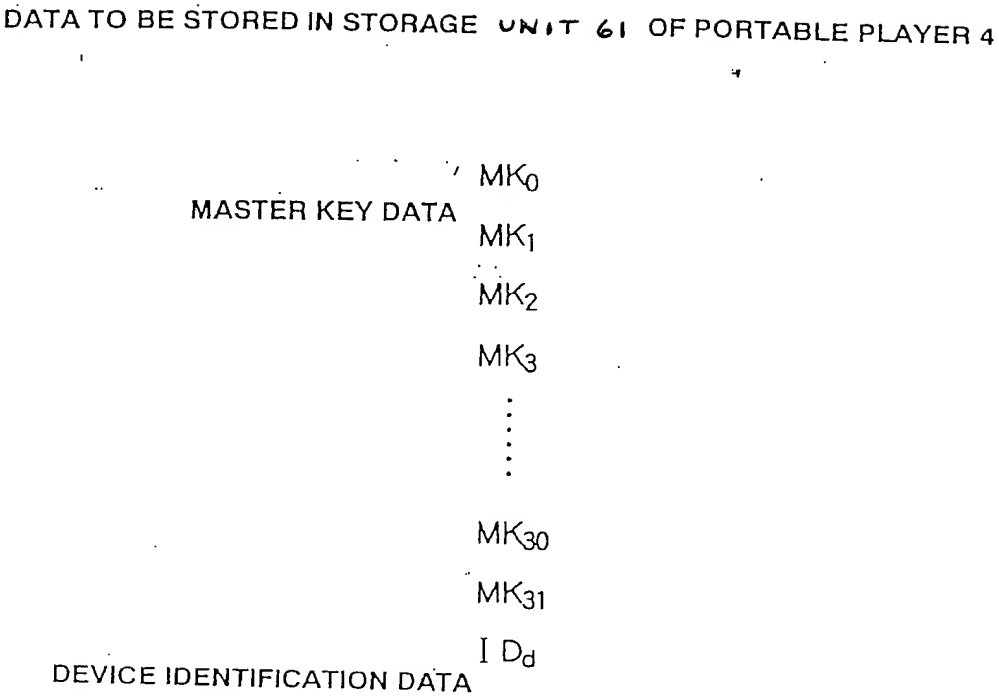

DATA TO BE STORED IN STORAGE UNIT 61 OF PORTABLE PLAYER 4
MASTER KEY DATA
MK0
MK1
MK2
MK3
MK30
MK31
I Dd
DEVICE IDENTIFICATION DATA

Figure 18

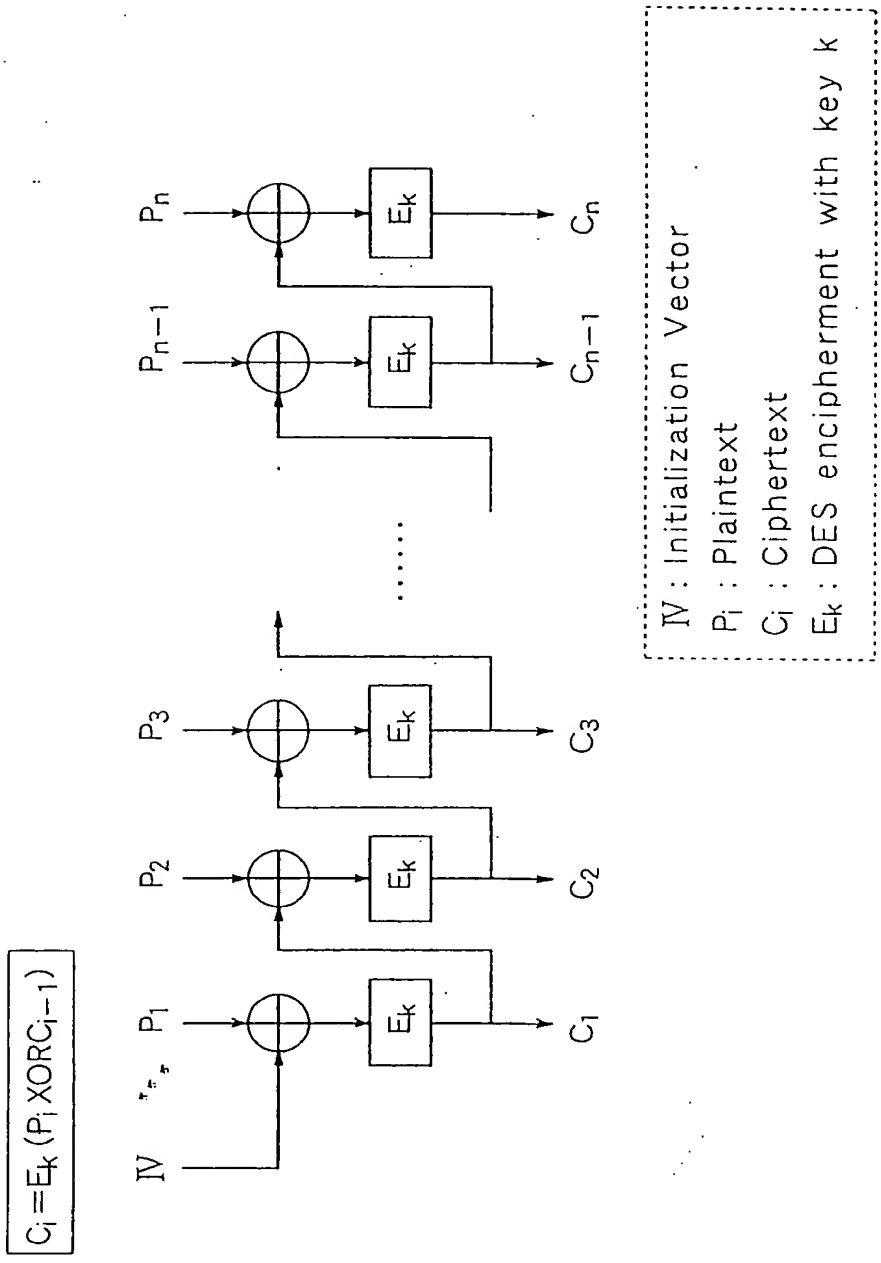

DES CBC MODE (ENCRYPTION)
$C_i = E_k(P_i \text{ XOR } C_{i-1})$
IV
$P_1$
$P_2$
$P_3$
$P_{n-1}$
$P_n$
$E_k$
$C_1$
$C_2$
$C_3$
$C_{n-1}$
$C_n$
IV : Initialization Vector
$P_i$ : Plaintext
$C_i$ : Ciphertext
$E_k$ : DES encipherment with key k

Figure 19

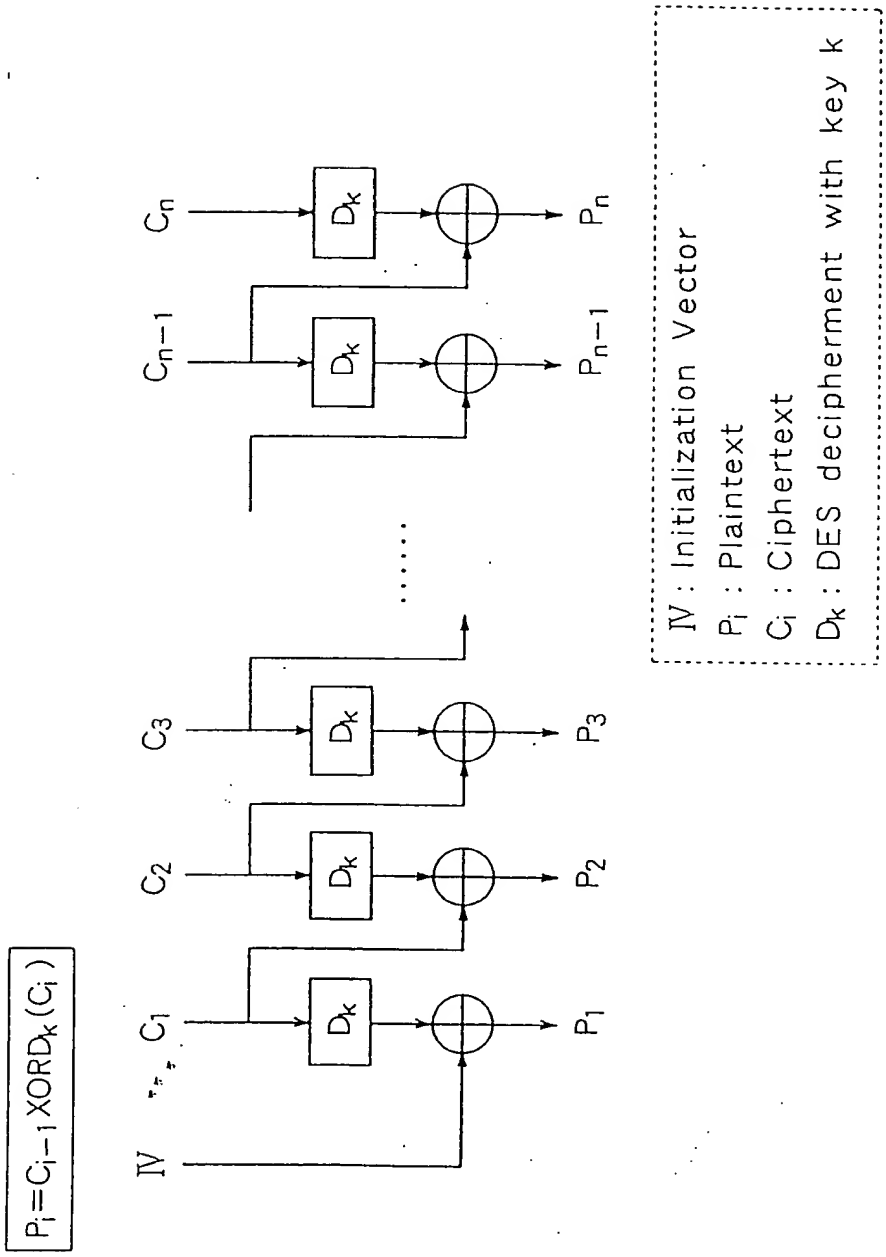
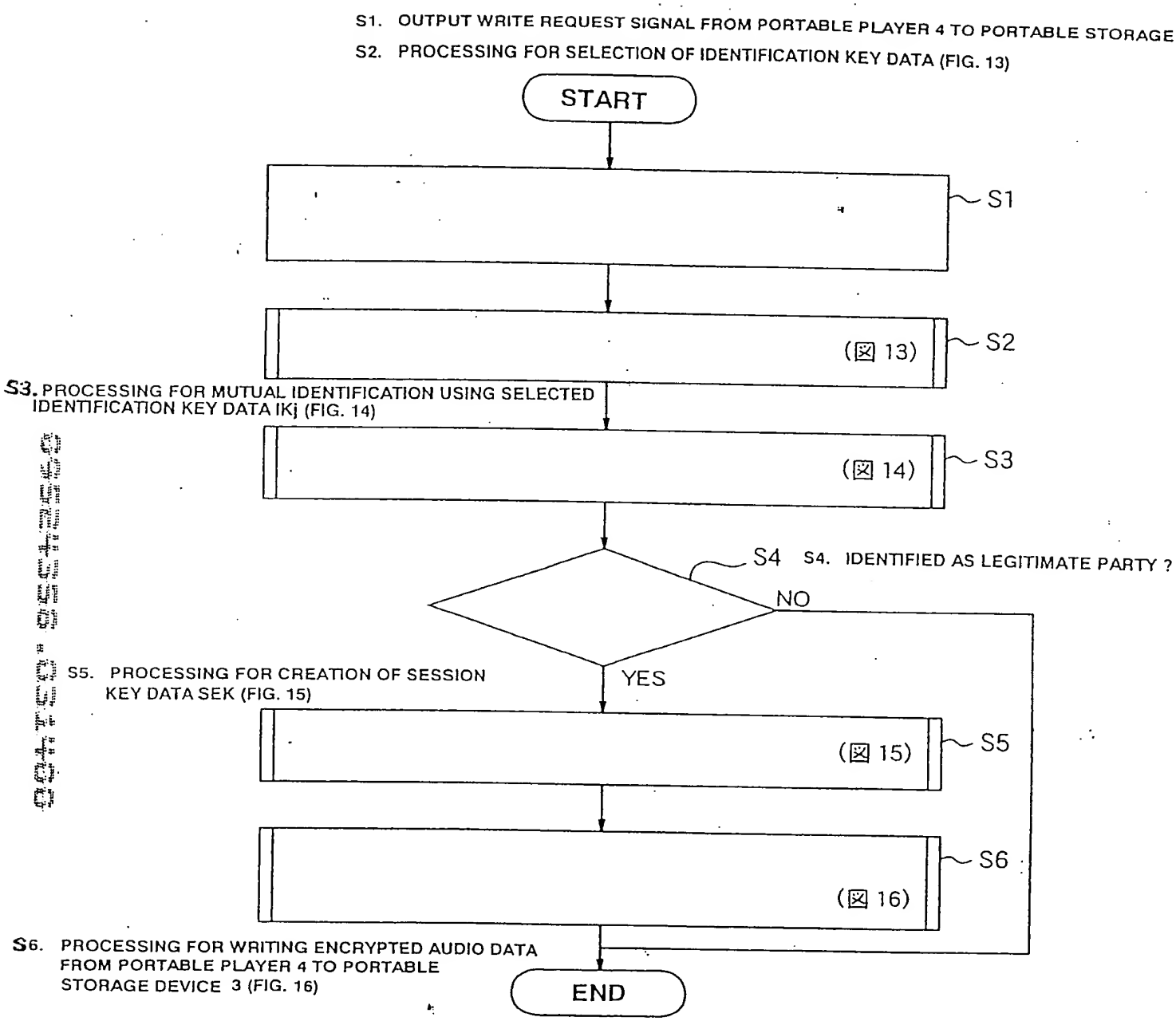
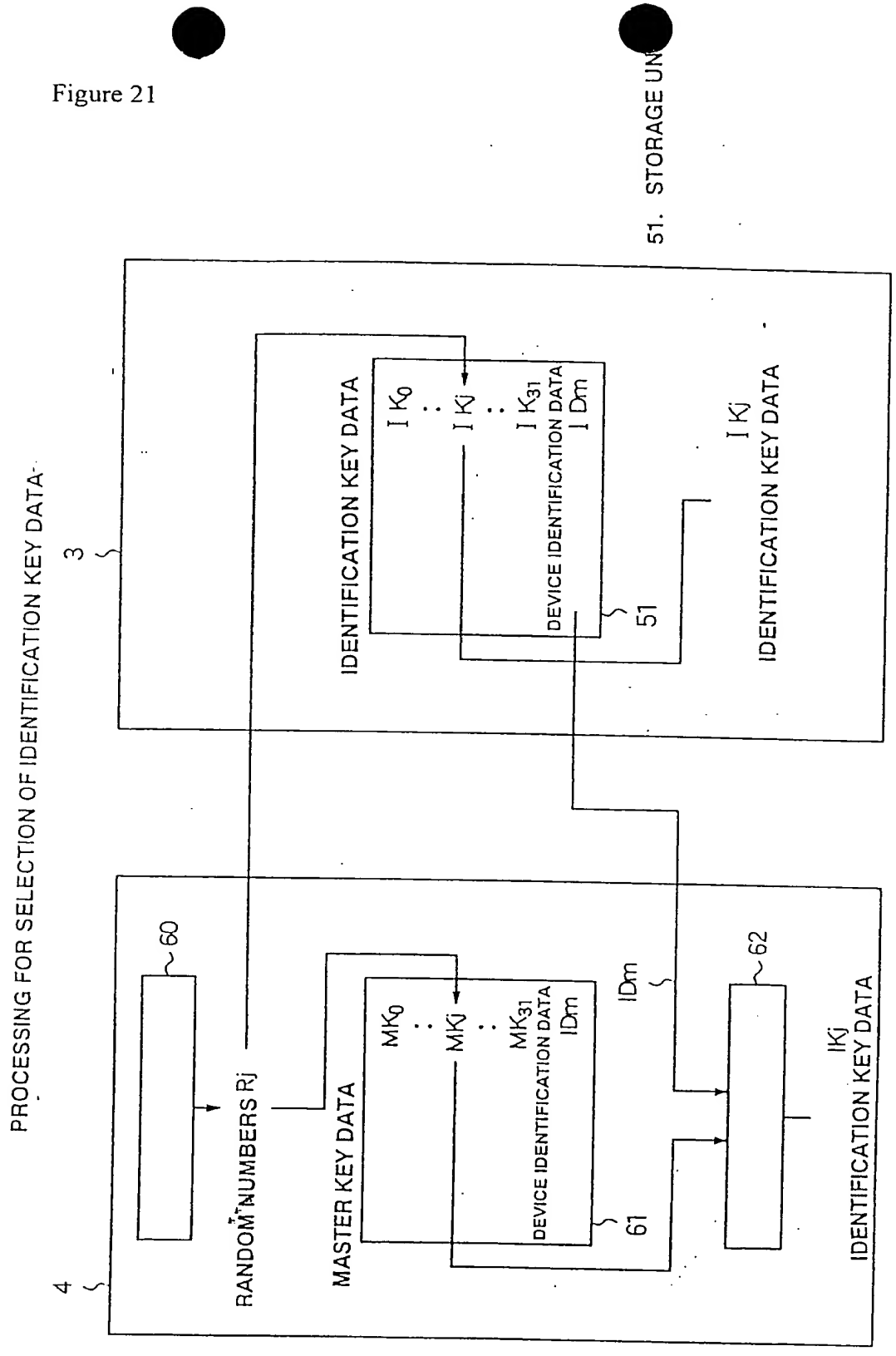

DES CBC MODE (DECRYPTION)
$P_i = C_{i-1} XOR D_k(C_i)$
IV
$C_1$
$C_2$
$C_3$
$C_{n-1}$
$C_n$
$D_k$
$P_1$
$P_2$
$P_3$
$P_{n-1}$
$P_n$
IV : Initialization Vector
$P_i$ : Plaintext
$C_i$ : Ciphertext
$D_k$ : DES decipherment with key k

Figure 20



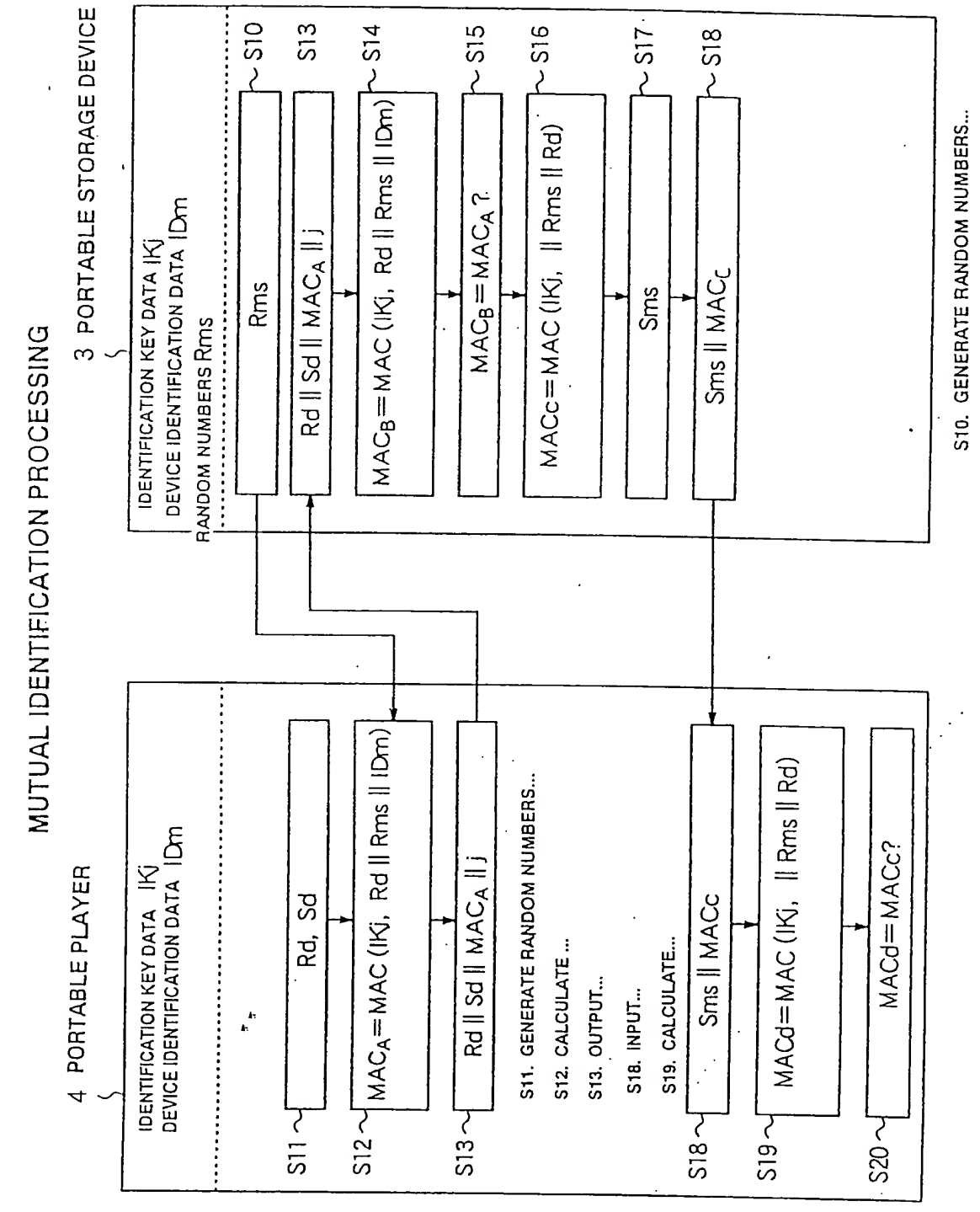

WRITE PROCESSING TO PORTABLE STORAGE DEVICE 3

Figure 21




60. RANDOM NUMBER GENERATION UNIT
61. STORAGE UNIT
62. KEY CREATION PROCESSING UNIT

Figure 22


MUTUAL IDENTIFICATION PROCESSING
4 PORTABLE PLAYER
IDENTIFICATION KEY DATA IKj
DEVICE IDENTIFICATION DATA IDm
3 PORTABLE STORAGE DEVICE
IDENTIFICATION KEY DATA IKj
DEVICE IDENTIFICATION DATA IDm
RANDOM NUMBERS Rms
S10
Rms
S13
Rd || Sd || MACA || j
S14
MACB = MAC (IKj, Rd || Rms || IDm)
S15
MACB = MACA ?
S16
MACc = MAC (IKj, || Rms || Rd)
S17
Sms
S18
Sms || MACc
S11
Rd, Sd
S12
MACA = MAC (IKj, Rd || Rms || IDm)
S13
Rd || Sd || MACA || j
S18
Sms || MACc
S19
MACd = MAC (IKj, || Rms || Rd)
S20
MACd = MACc?
S11. GENERATE RANDOM NUMBERS...
S12. CALCULATE ...
S13. OUTPUT...
S18. INPUT...
S19. CALCULATE...
S10. GENERATE RANDOM NUMBERS...
S13. INPUT...
S14. CALCULATE...
S16. CALCULATE...
S17. GENERATE RANDOM NUMBERS...

PROCESSING FOR CREATION OF SESSION KEY DATA

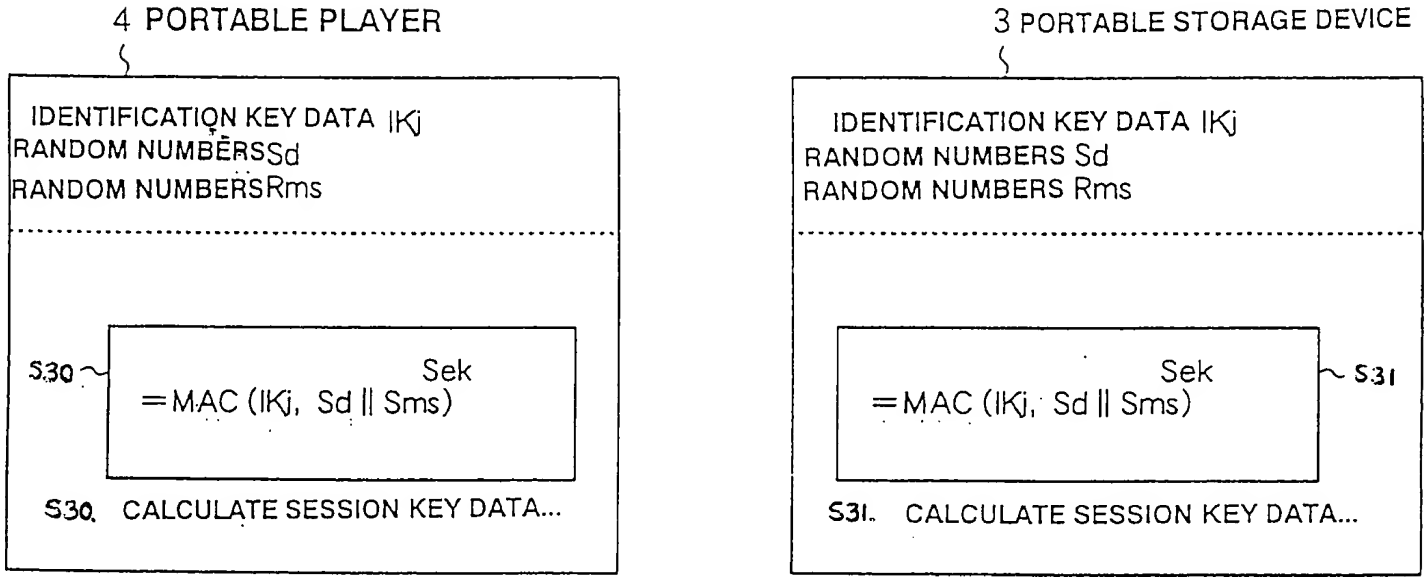



Figure 23

Figure 24

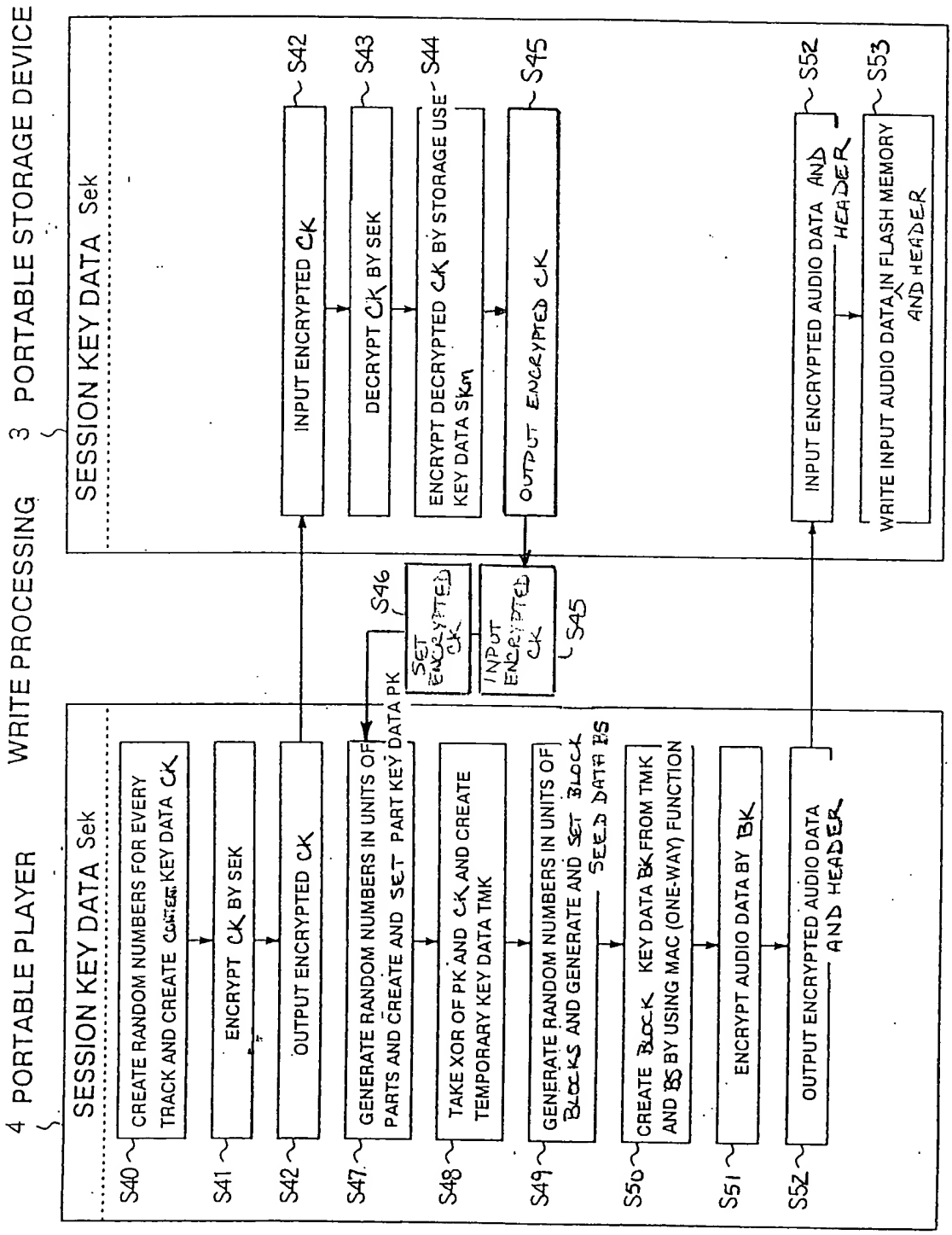
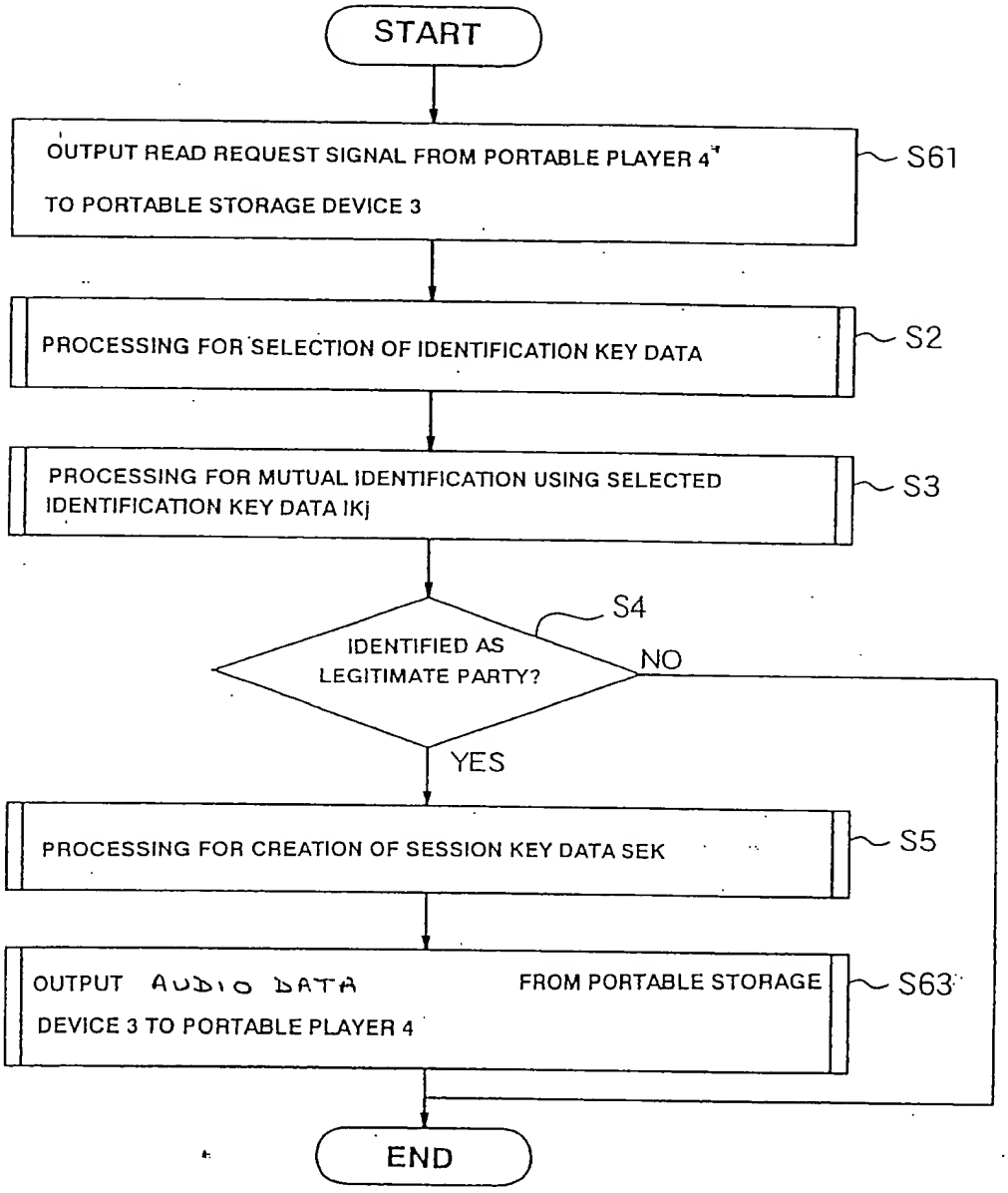

4 PORTABLE PLAYER
WRITE PROCESSING
3 PORTABLE STORAGE DEVICE
SESSION KEY DATA Sek
SESSION KEY DATA Sek
CREATE RANDOM NUMBERS FOR EVERY TRACK AND CREATE CONTENT KEY DATA CK
ENCRYPT CK BY SEK
OUTPUT ENCRYPTED CK
GENERATE RANDOM NUMBERS IN UNITS OF PARTS AND CREATE AND SET PART KEY DATA PK
TAKE XOR OF PK AND CK AND CREATE TEMPORARY KEY DATA TMK
GENERATE RANDOM NUMBERS IN UNITS OF BLOCKS AND GENERATE AND SET BLOCK SEED DATA BS
CREATE BLOCK KEY DATA BK FROM TMK AND BS BY USING MAC (ONE-WAY) FUNCTION
ENCRYPT AUDIO DATA BY BK
OUTPUT ENCRYPTED AUDIO DATA AND HEADER
INPUT ENCRYPTED CK
DECRYPT CK BY SEK
ENCRYPT DECRYPTED CK BY STORAGE USE KEY DATA Skm
OUTPUT ENCRYPTED CK
SET ENCRYPTED CK
INPUT ENCRYPTED CK
INPUT ENCRYPTED AUDIO DATA AND HEADER
WRITE INPUT AUDIO DATA IN FLASH MEMORY AND HEADER
S40
S41
S42
S47
S48
S49
S50
S51
S52
S42
S43
S44
S45
S46
S45
S52
S53

Figure 25




PROCESSING FOR READING FROM PORTABLE STORAGE DEVICE 3

Figure 26

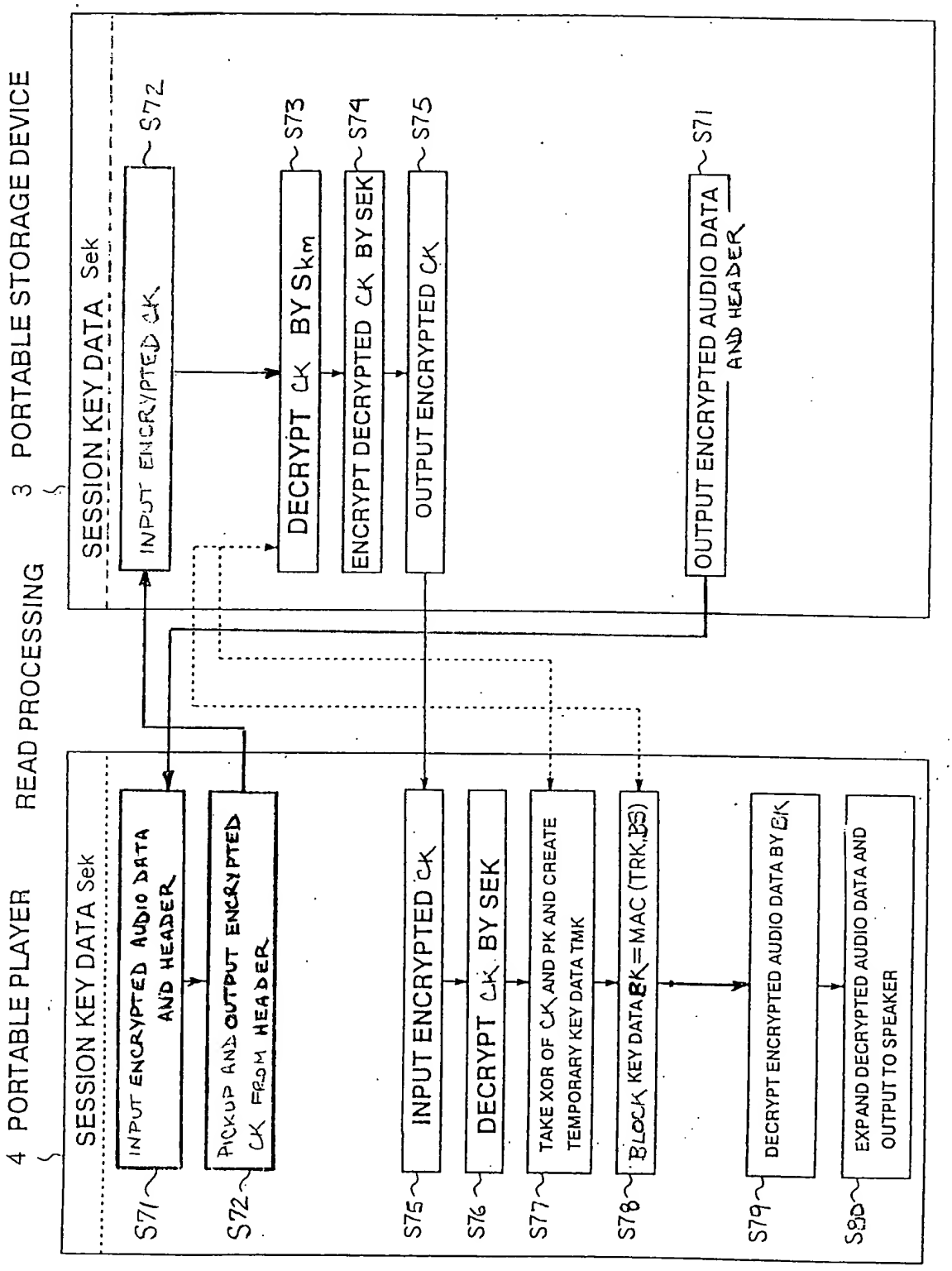

3 PORTABLE STORAGE DEVICE
SESSION KEY DATA Sek
INPUT ENCRYPTED CK
S72
DECRYPT CK BY Skm
S73
ENCRYPT DECRYPTED CK BY SEK
S74
OUTPUT ENCRYPTED CK
S75
OUTPUT ENCRYPTED AUDIO DATA AND HEADER
S71
READ PROCESSING
4 PORTABLE PLAYER
SESSION KEY DATA Sek
INPUT ENCRYPTED AUDIO DATA AND HEADER
S71
PICKUP AND OUTPUT ENCRYPTED CK FROM HEADER
S72
INPUT ENCRYPTED CK
S75
DECRYPT CK BY SEK
S76
TAKE XOR OF CK AND PK AND CREATE TEMPORARY KEY DATA TMK
S77
BLOCK KEY DATA BK = MAC (TRK, BS)
S78
DECRYPT ENCRYPTED AUDIO DATA BY CK
S79
EXPAND DECRYPTED AUDIO DATA AND OUTPUT TO SPEAKER
S80

Figure 27

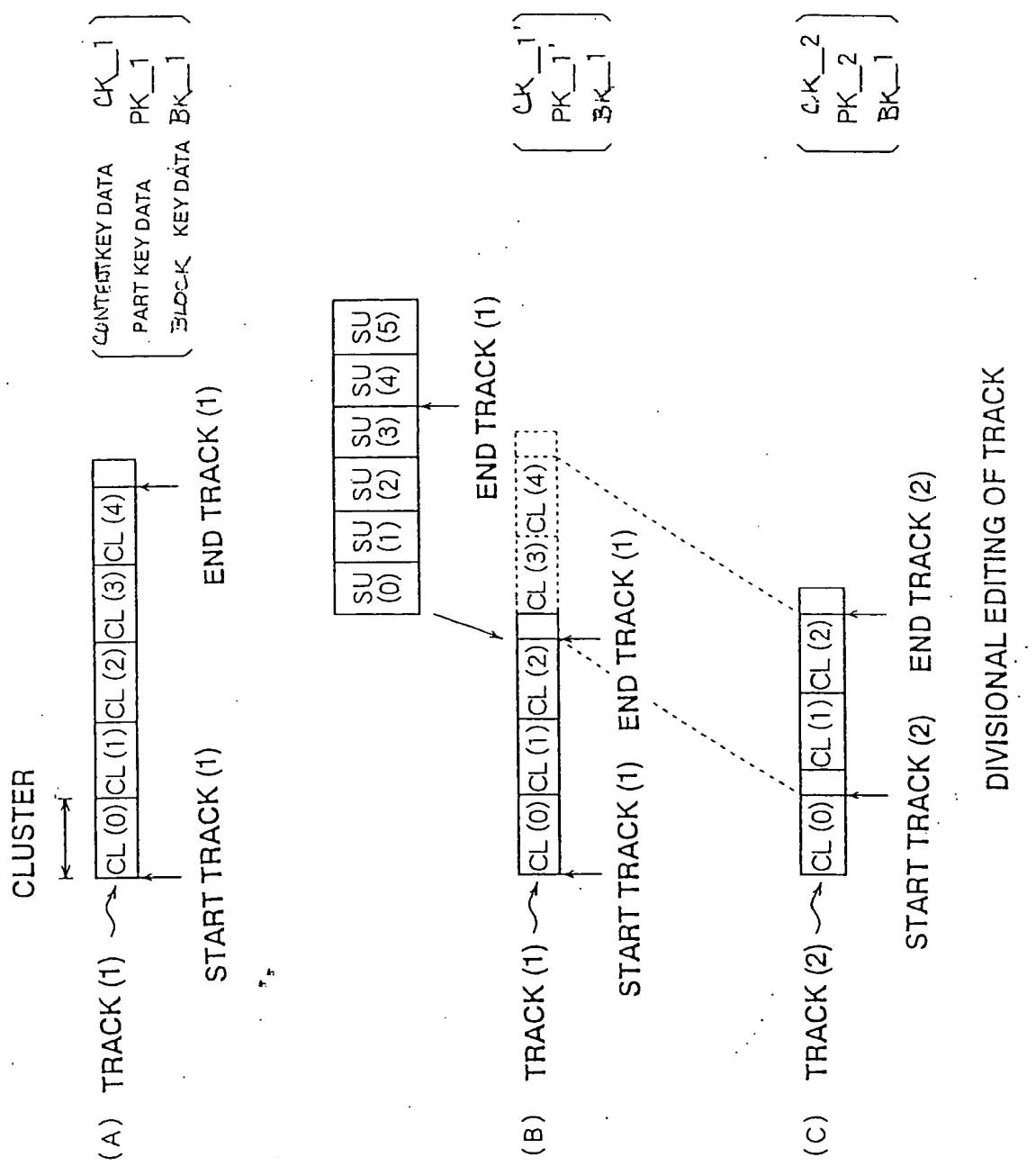

CLUSTER
(A) TRACK (1)
CL (0) CL (1) CL (2) CL (3) CL (4)
START TRACK (1)
END TRACK (1)
CONTENT KEY DATA CK_1
PART KEY DATA PK_1
BLOCK KEY DATA BK_1
SU (0) SU (1) SU (2) SU (3) SU (4) SU (5)
END TRACK (1)
(B) TRACK (1)
CL (0) CL (1) CL (2) CL (3) CL (4)
START TRACK (1)
END TRACK (1)
CK_1'
PK_1'
BK_1
(C) TRACK (2)
CL (0) CL (1) CL (2)
START TRACK (2)
END TRACK (2)
CK_2
PK_2
BK_1
DIVISIONAL EDITING OF TRACK

Figure 28

Cluster CL(2) of Track(1) after divide

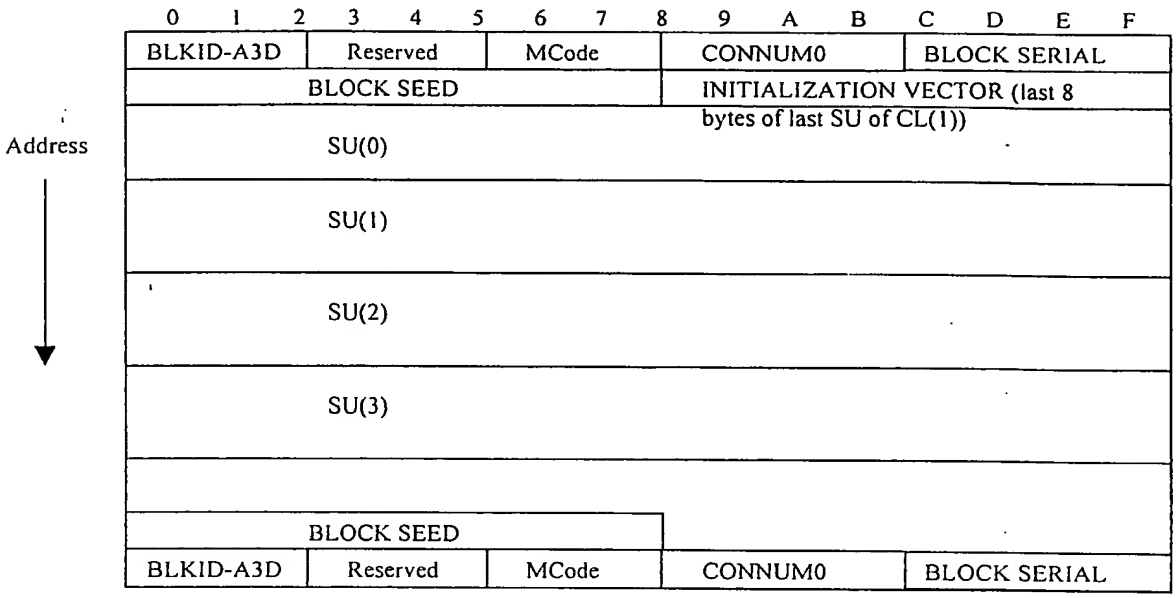



Figure 29

Cluster CL(0) of Track(2) after divide

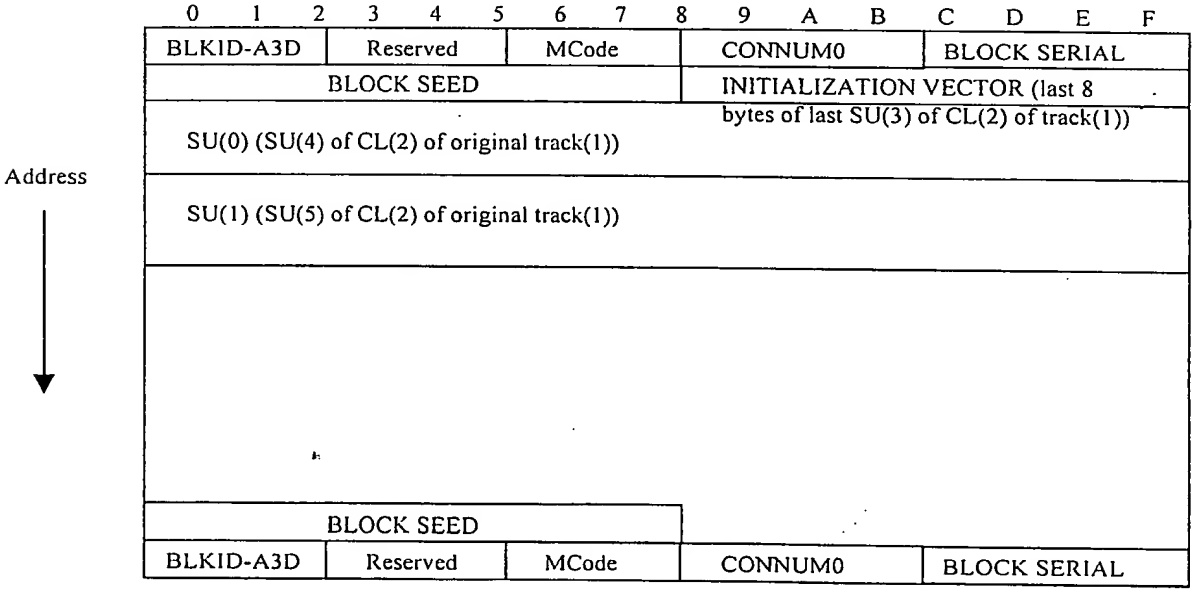
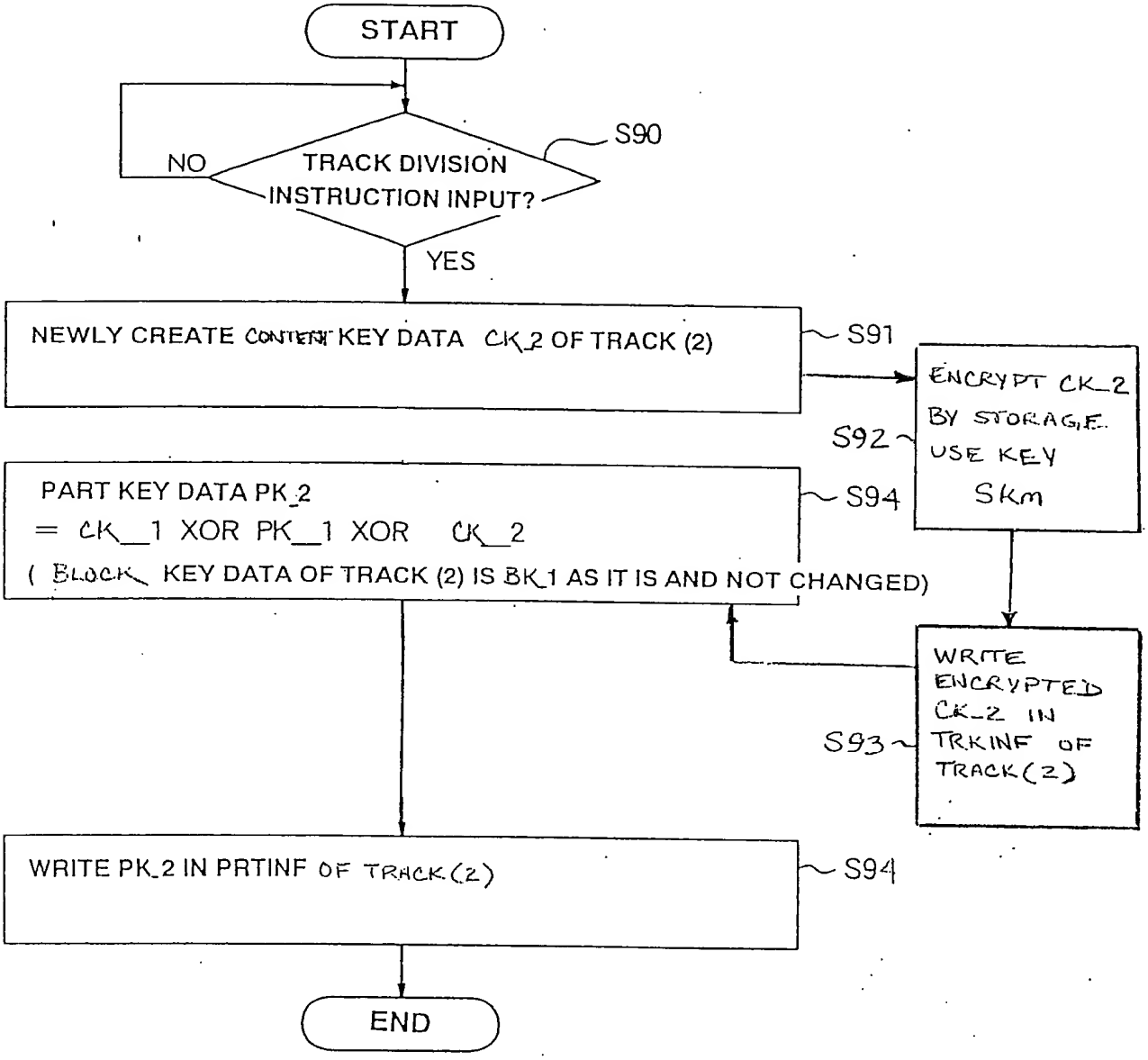
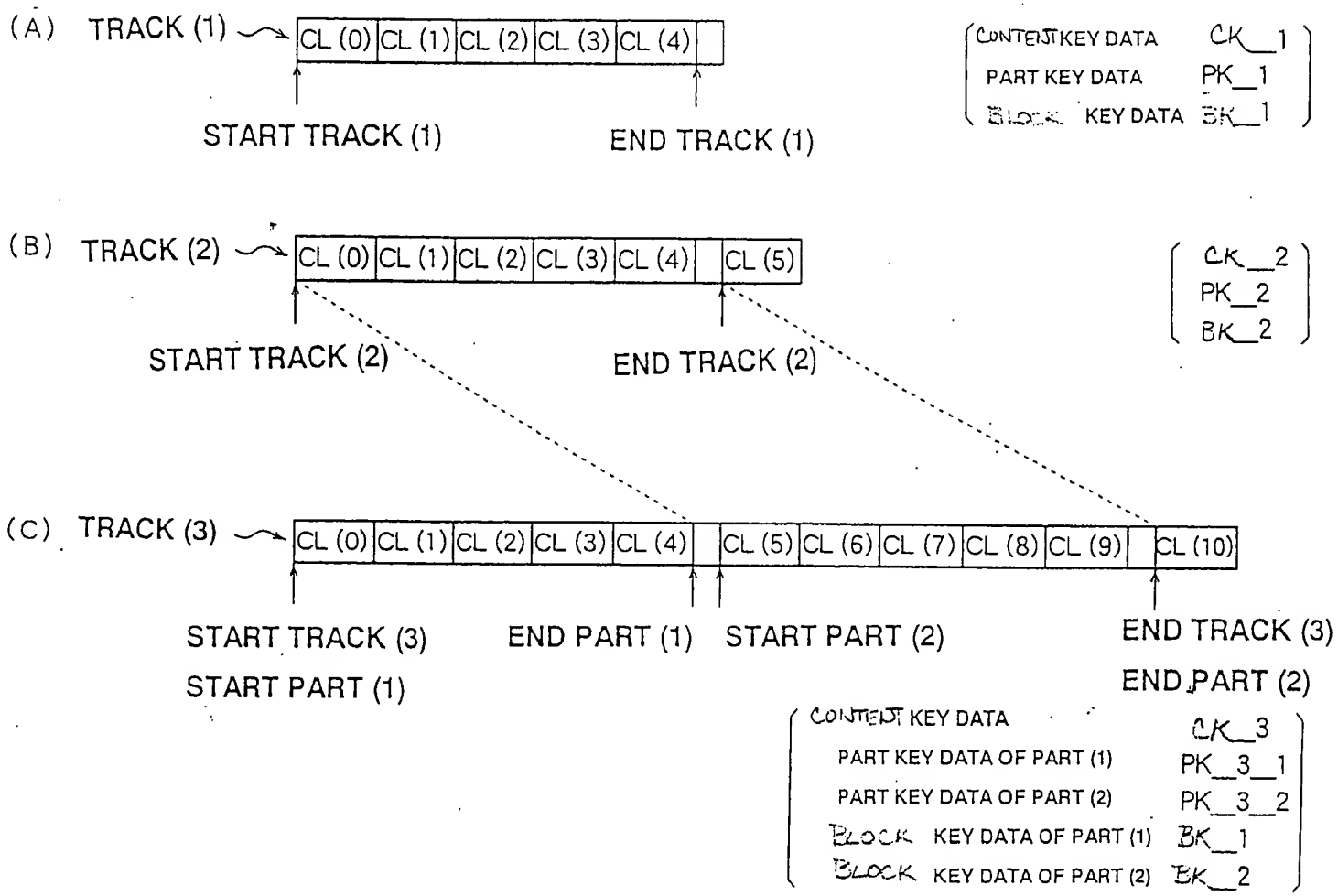


Figure 30

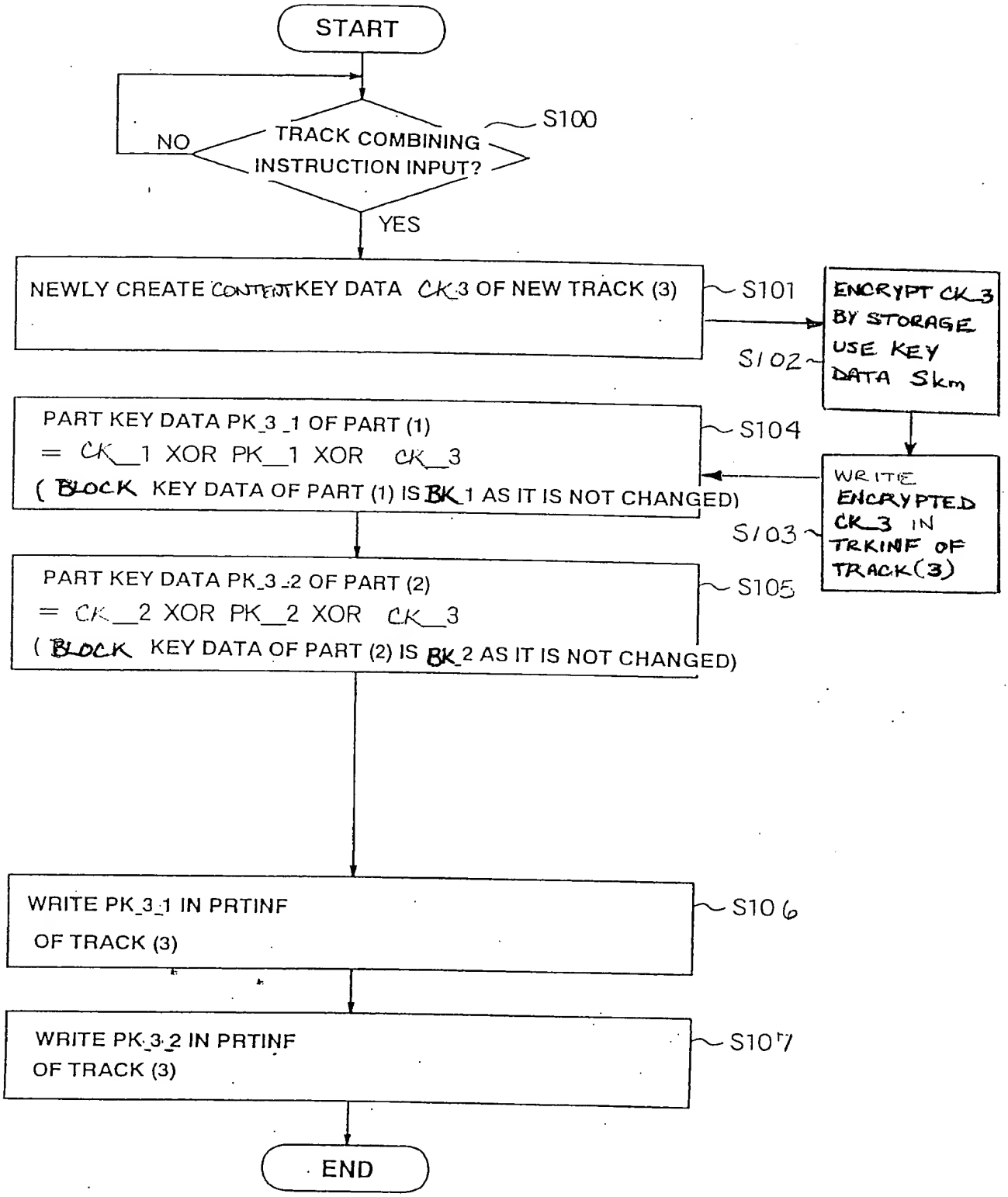

START
S90
NO
TRACK DIVISION INSTRUCTION INPUT?
YES
NEWLY CREATE CONTENT KEY DATA CK_2 OF TRACK (2)
S91
ENCRYPT CK_2 BY STORAGE USE KEY Skm
S92
PART KEY DATA PK_2 = CK__1 XOR PK__1 XOR CK__2
( BLOCK KEY DATA OF TRACK (2) IS BK_1 AS IT IS AND NOT CHANGED)
S94
WRITE ENCRYPTED CK_2 IN TRKINF OF TRACK(2)
S93
WRITE PK_2 IN PRTINF OF TRACK(2)
S94
END




TRACK COUPLING PROCESSING

Figure 31

Figure 32


START
TRACK COMBINING INSTRUCTION INPUT?
S100
NO
YES
NEWLY CREATE CONTENT KEY DATA CK_3 OF NEW TRACK (3)
S101
ENCRYPT CK_3 BY STORAGE USE KEY DATA Skm
S102
WRITE ENCRYPTED CK_3 IN TRKINF OF TRACK(3)
S103
PART KEY DATA PK_3_1 OF PART (1)
= CK_1 XOR PK_1 XOR CK_3
( BLOCK KEY DATA OF PART (1) IS BK_1 AS IT IS NOT CHANGED)
S104
PART KEY DATA PK_3_2 OF PART (2)
= CK_2 XOR PK_2 XOR CK_3
( BLOCK KEY DATA OF PART (2) IS BK_2 AS IT IS NOT CHANGED)
S105
WRITE PK_3_1 IN PRTINF OF TRACK (3)
S106
WRITE PK_3_2 IN PRTINF OF TRACK (3)
S107
END